Dynamical Code による unfolding Color Code

Fu と Gottesman の Error Correction in Dynamical Codes(Dynamical Code はおそらく Subsystem と Floquet Code の総称) [2] の前半部分を参考にして、Gidney の Magic State Cultivation [3] の Color Code から Surface Code への escape を考えてみた。

1. Dynamical Code による Color Code の unfolding (Color Code → Surface Code)

ここでは、Fig.1 のような符号距離 5 の Color Code について考える。

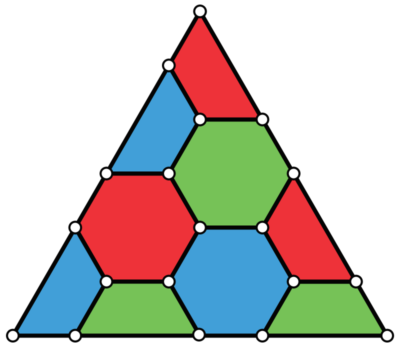

Fig. 1

最初に Color Code の X logical operator を green boundary に沿う 5-weight operator、Z logaical operator を red boundary に沿う 5-weight operator と定義する。ここでは Color Code を Fig.1 の blue boundary を折り目として開くことを考える。Color Code を unfold する直前は Fig.2 のように折り目の boudary 側に 0 初期化された量子ビットを、unfold する Color Code の量子ビット数 n から符号距離 d を引いた $n-d$ 個だけ用意する。

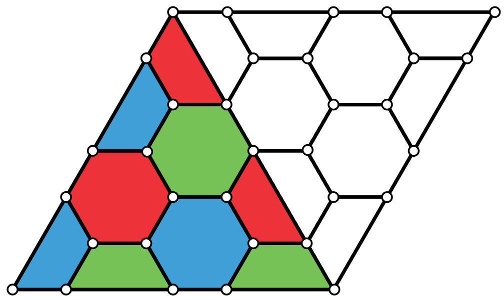

Fig. 2

ここでスタビライザー群を $\mathcal{S}$ と表す。Fig.3 のように green の edge に X edge stabilizer、boundary 上の blue edge に対して Z edge stabilizer を $\mathcal{S}$ に追加する。ただし、追加された量子ビットの領域に対しても Color Code が続いているように見て、edge operator を定義している。これによって、red face の Z stabilizer は $\mathcal{S}$ から消える。またそれと同時に、green face の X stabilizer、red face の Z stabilizer、blue face の Z stabilizer のシンドローム測定を、追加した量子ビットの領域で開始する。そうすると、Fig.3 のようになる。

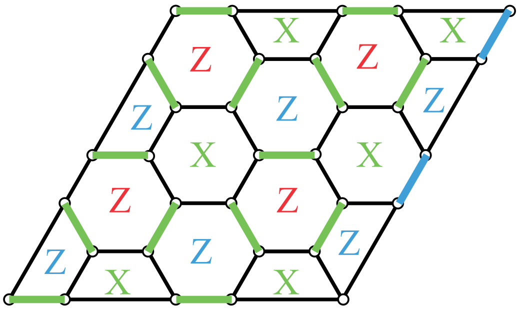

Fig. 3

ということで変換後のスタビライザー $\mathcal{S}$ は Surface Code に対応するものとなっており、Fig.4 に示すようになっている。ただし、橙色が X stabilizer、青色が Z stabilizer を表す。ここで定義している 6-weight もしくは 4-weight のスタビライザーは floquet code 的に実装するのではなく、surface code のように実際の weight 分のシンドローム測定を行う。

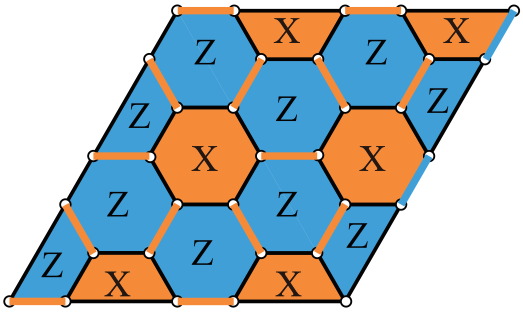

Fig. 4

ここで、最初に定義した logical operator は追加したすべてのスタビライザーと可換であるため論理情報は保存される ([2] を参照)。

2. 拡大

　ここまでで、Surface Code にできたのであとは拡大するだけである。拡大は Fowler と Gidney の [1] の Fig.11 と同じようにすればできると思う (Fig.5)。

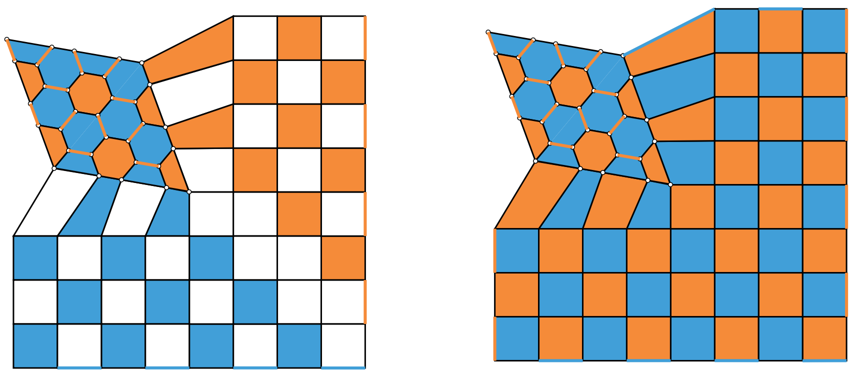

Fig. 5

またここまでの操作はいっぺんにやってしまえば良いので、Fig.6 のように、拡大先の Surface Code の量子ビットをすべて用意しておいて、その後全体で Surface Code のシンドローム測定をすることによって、escape stage が完成すると思う。

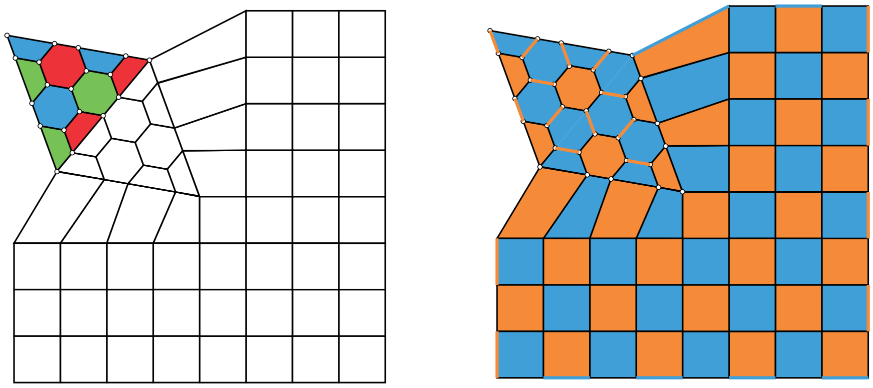

Fig. 6

3. 検証

ここでは、検証を簡単にするために符号距離 3 の場合について考える。まずは Color Code から Surface Code への変換がエラーが無いときに所望の変換になっているかを検証する。ただし、スタビライザーの測定結果はすべて + が出ることを仮定している。Fig.7 より、Color Code にエンコードし、Surface Code に変換する場合と、最初から Surface Code に変換する場合で Logical Operator と Stabilizer が同じであることがわかる。よって、Color Code から Surface Code への変換は所望の操作になっている。

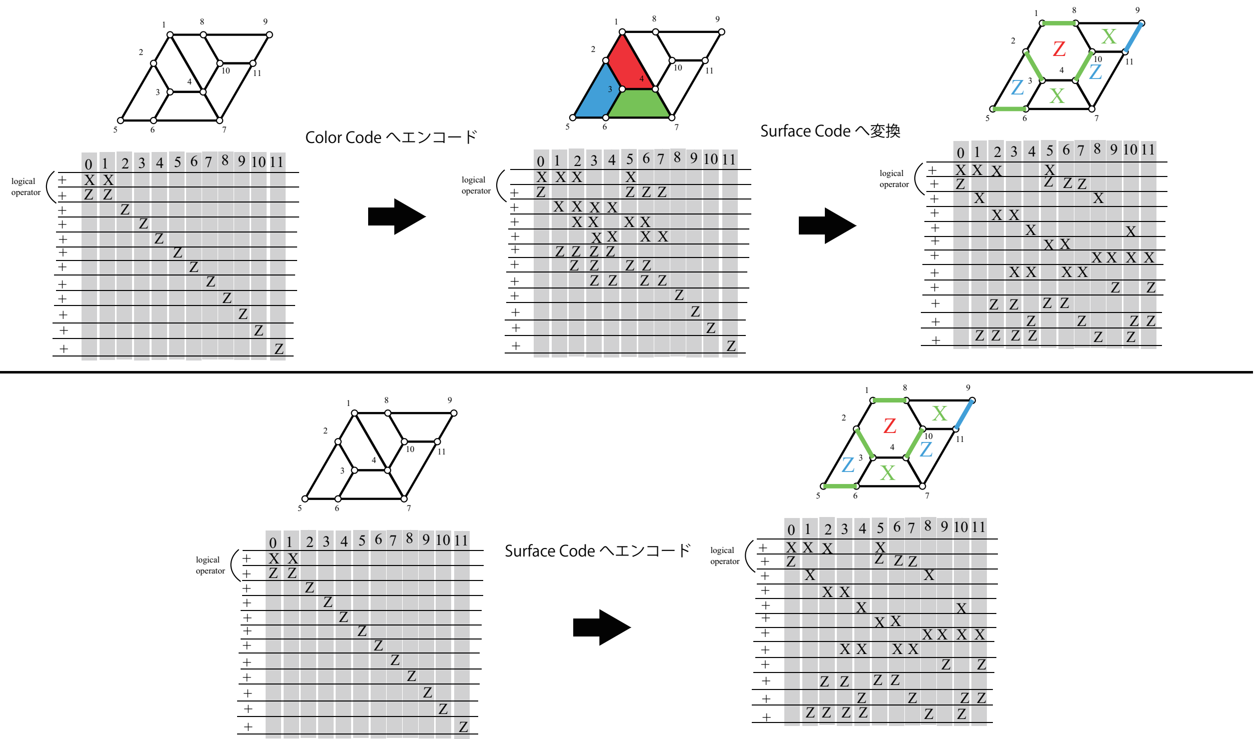

Fig. 7

また、拡大操作を検証すると、Fig.8 のようになる。Fig.8 の上段は Fig.7 の最終状態から、拡大する操作を表し、下段は拡大された Surface Code にエンコードする操作を表す。上段と下段の最終状態は等しいので、拡大操作は所望の操作になっている。

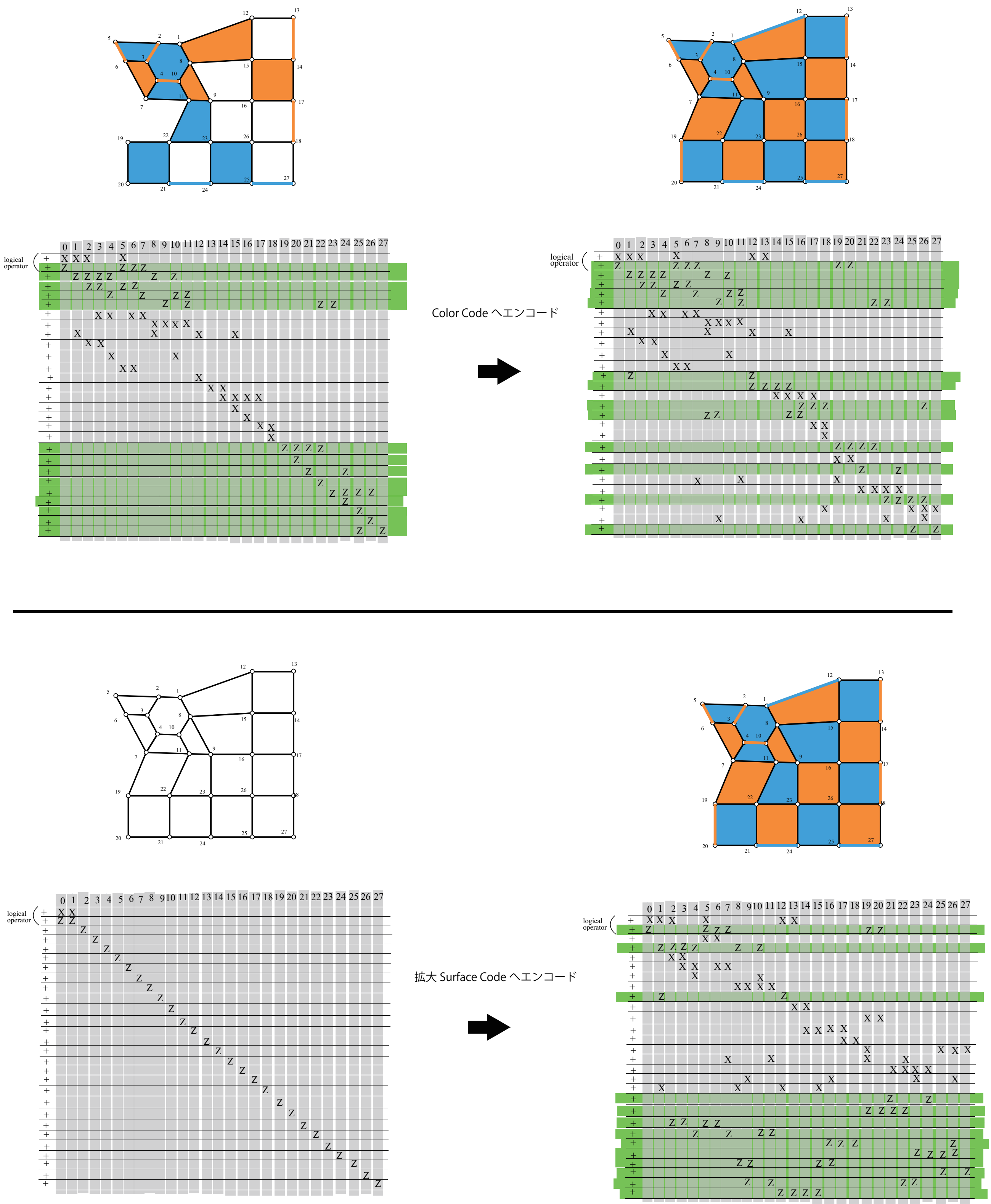

Fig. 8

最後にこれらの操作がいっぺんにできるかを検証する。Fig.9 に示す通り、Color Code からいっぺんに変換、拡大の操作を行っても、Fig.8 の下段と同じ状態が得られることがわかる。

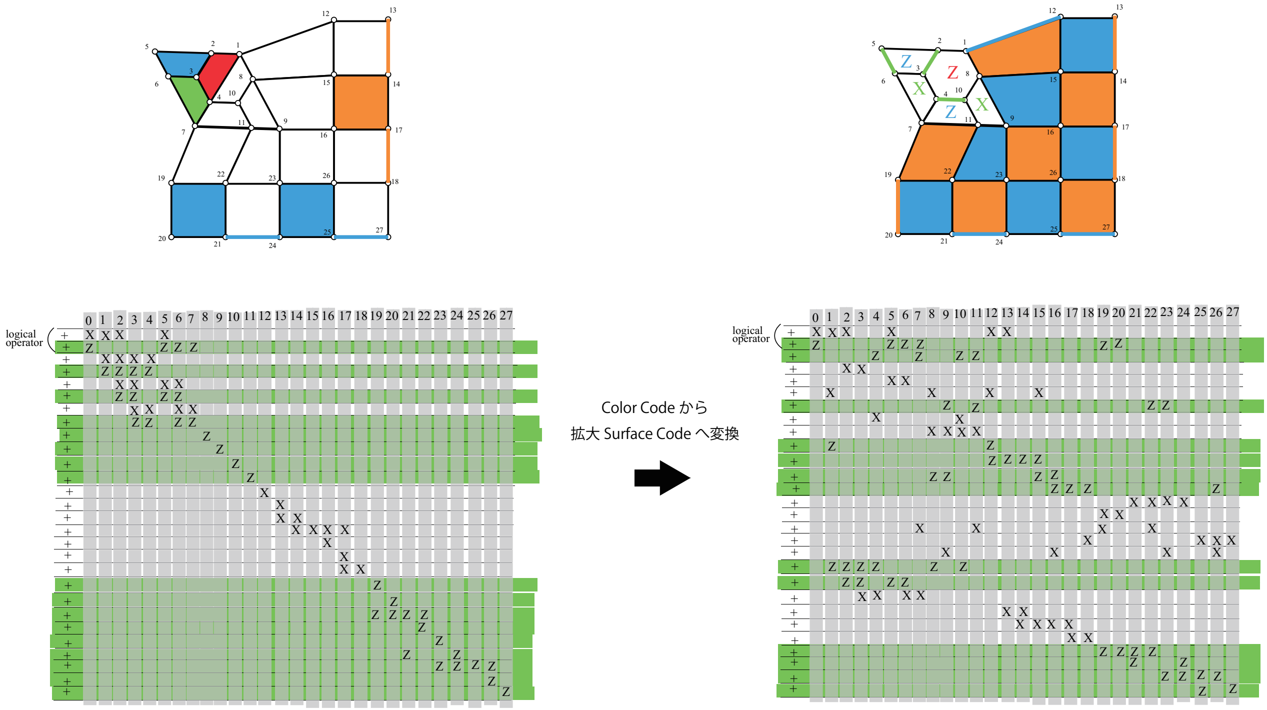

Fig. 9

REFERENCES

[1] Austin G. Fowler and Craig Gidney. "Low overhead quantum computation using lattice surgery", arXiv preprint. eprint: `1908.06709`. URL: `https://arxiv.org/abs/1908.06709`.

[2] Xiaozhen Fu and Daniel Gottesman. "Error Correction in Dynamical Codes", arXiv preprint. eprint: `2403.04163`. URL: `https://arxiv.org/abs/2403.04163`.

[3] Craig Gidney, Noah Shutty, and Cody Jones. "Magic state cultivation: growing T states as cheap as CNOT gates", arXiv preprint. eprint: `2409.17595`. URL: `https://arxiv.org/abs/2409.17595`.

Update rules from [2]

Lemma 1 (**Stabilizer Update Rules**)
Let $\mathcal{S}$ be the stabilizer generators with a stabilizer state $|\psi\rangle$ being either in $+1$ or -1 eigenstate of the generators.
Let m be a Pauli measurement performed on $|\psi\rangle$, and denote the outcome of m by $O(m) \in \{\pm 1\}$.

1. If $\pm m \in \langle \mathcal{S} \rangle$, then the outcome is fixed by the eigenvalues of stabilizers for $|\psi\rangle$, and the state remains unchanged.
2. If m anti-commutes with some elements in $\mathcal{S}$: Let $V = \{s_1, s_2, \cdots, s_l\}$ be a subset of $\mathcal{S}$ whose elements anti-commute with m. We replace s_1 with m and update the rest of V by $s_i \to s_i \cdot s_1$, for $2 \leq i \leq l$. $\mathcal{S} \cap V$ is now updated to $\{O(m) \cdot m, s_2 \cdot s_1, s_3 \cdot s_1, \cdots, s_l \cdot s_1\}$
3. If $\pm m \notin \langle \mathcal{S} \rangle$ and $[m, s] = 0 \, \forall s \in \mathcal{S}$, then we update the set of stabilizer generators: $\mathcal{S} \to \mathcal{S} \cup \{O(m) \cdot m\}$. This assumes that m is not a logical operator.

Lemma 2 (**Logical Update Rules**)
Let L be a logical operator of a stabilizer group $\langle S \rangle$ and let $|\psi\rangle$ be an eigenstate of L.
Let m be a Pauli measurement performed on $|\psi\rangle$ and denote the outcome by $O(m) \in \{\pm 1\}$.

1. If $m = (-1)^a \cdot L$, then $O(m) \cdot (-1)^a$ gives the eigenvalue of L for the state $|\psi\rangle$, and the logical operator remains unchanged.
2. If m commutes with L, the logical operator remains unchanged.
3. If m anti-commutes with L and commutes with $\langle S \rangle$, then L is updated to $O(m) \cdot m$. The new state is a $+1$ eigenstate of $O(m) \cdot m$ instead of L.
4. If m anti-commutes with L and anti-commutes with some elements in S: In the stabilizer update rules, we replace an element s_1 with m and update the rest of the elements in S that anti-commute with m using the 2^{nd} rule in Lemma 1. For the logical operator, we update $L \to L \cdot s_1$, where s_1 is the element that is replaced with m.